dave
nutting
associates

a division of bally manufacturing corporation

527 west golf road
arlington heights.
illinois 60005
(312) 956-0710

ASTROVISION BASIC SUMMARY

Editing feature

An edit key has been provided. To use type the line number of
a line already in the computer, then type the sequence WORD-SPACE.
This will draw from the stored line one character or token as if you
typed it yourself. By repeating WORD-SPACE keystrokes you can step thru
the stored line to the point where correction is to be made. New
characters and erasures can be interspersed at any time. If you run
off the end of the stored line a GO character will be entered, storing
the updated line in memory.

List feature

Pressing the key in the upper right hand corner of the keypad
(divide, or LIST) while the program is running will cause the computer
to LIST each line before executing it. This allows the programmer to
trace the flow of statement execution.

Line command changed

The line command will no longer reject lines that start or stop
off the screen. Instead it will try to draw the portion that is visable.
This does not work perfectly.

"Bally Basic" no longer printed on reset, halt or error. This saves
space in the ROM and on the screen.

*() array added

Many users have complained about the behavior of @(). Presently
the contents of this array are altered as the programmer adds and
deletes program lines.
This happens because @() starts where the program storage area
ends, and no effort is made to relocate this block as the program
shrinks and expands.
The solution I have chosen was to add a second array, *() which
starts at the top of memory and works down. This worked out to require
less code than would be needed to relocate @(), and simplifies programs
that use two array areas.
These arrays do overlap each other. Low @() indexes correspond to
high *() indexes.
The diagram on the next page shows these relationships.

PROGRAM STORAGE START

PROGRAM

@ (∅)   * (SZ÷2)   @ ARRAY START

@ (SZ÷2)   * (∅)   * ARRAY START
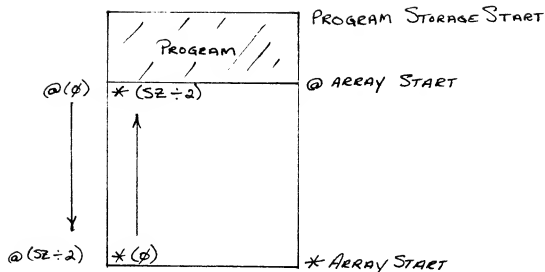
Tape commands

The integration of the audio cassette interface into the BASIC
ROM CARTRIDGE with the attendant increase in BAUD RATE has forced a
change in the commands and methods of tape use.
It is no longer possible to provide line by line feedback of tape
storage and loading.

Program write & read

To write a program onto tape, enter the :PRINT command. This will
write a leader, all the program statements, and trailer onto audio tape.
This will take 16 seconds or less to accomplish.

To load a program, reset the computer and type :INPUT. You will
notice a blue image in the background providing a binary display of the
loading process. If a checksum error is detected a question mark will
be typed out. Note that any existing program statements in the computer
before :INPUT will be forgotten.
Since the :INPUT command loads a binary ram image, a checksum
error on load could mean that critical control information was
corrupted. This means that an erroneous load may be impossible to
edit or execute, or could even "bomb the computer"

Data write &read

Special forms of :INPUT and :PRINT are recognized for reading and writing blocks of data under program control.

For example, to save the first 100 words of the @ array we would use:

    :PRINT @(0), 100

To load that block back into memory we could use:

    :INPUT @(0)

Since the * array works backwards to write the first 50 words of * out we must give the last address, instead of the first:

    :PRINT *(49), 50

(this writes *(49) to *(0) out)

similar shenanigans to laod * :

    :INPUT *(49)

Again, a checksum error will print a question mark.

Finally, the :RUN command is still available for bootstrapping machine language programs. :RUN will load starting at 4000 hex a block of variable size. When loaded control will be transfered to 4000 hex.